2G3-6

モジュラリティの差異に基づくコントラスト法

Patterns with Emerging Modularity and Their Detection

鶴田 哲章　　原口 誠
Noriaki Tsuruta　Makoto HARAGUCHI

北海道大学大学院情報科学研究科コンピュータサイエンス専攻
Division of Computer Science, Graduate School of Information Science and Technology, Hokkaido University

This paper presents a simple but effective method for contrasting the degree of connectivity among terms over several document-term databases and for finding patterns with "emerging connectivity". In order to consider potential power for terms to be connected each other, we use the Newman's modularity notion. Thus we can say that the targets are patterns with "emerging modularity". The emergingness is simply measured by subtracting the modularity in one database with one in another database. So we can regard the problem of mining patterns with emerging modularity as a kind of optimization problem to maximize the subtraction under some constraint to exclude extreme cases. In fact, such an optimization algorithm is presented based on branch-and-bound techniques and is showed its usefulness by some experiments.

1. はじめに

データマイニング手法に関する主要なテーマとして、**頻出アイテム集合発見問題**[2] が注目され久しい。この研究では、**相関ルール**[2] や相関しているアイテムセットマイニングに基づく頻出パターン発見や、**Emerging pattern**[3] を発見する研究では、与えられたデータベース、あるいは与えられた幾つかのデータベースのうちの 1 つのデータベースにおいて顕著に高い生起頻度が見られるパターン抽出の手法が多く研究されている。

そのような「頻度が高い」といった特徴を持ったものは重要であるという認識は、重要度を測る上での一つの有効な経験則であるが一方で、頻度や相関がそれほど高くはないものの中にも重要なものが潜む可能性があると、本研究では考える。

本研究では、非頻出なパターンの中でも「度のデータベースでも頻度は低いが、必然性 (モジュラリティ) が変化 (増加) するパターン」を重要な概念と考え、抽出を試みる。**モジュラリティ**[1] とは「偶発的な結び付きとの差異」という意味での必然性を示したものであり、多くの他のアイテムとの結び付きを持つアイテム同士の結合は「偶発的に結びつく可能性が高い (必然性の低い結合)」として低い評価を付与し、逆に他のアイテムとの結び付きが少ないアイテム同士の結合は「有意な結合 (必然性の高い結合)」として高く評価する。ランダムグラフとの差を観察しているわけである。この「モジュラリティが顕在化するパターン」とは、伝統的なデータマイニングの研究での抽出目標である頻出パターンではなく、パターンのモジュラリティが対象とするデータベースでは低いが、それとは別のデータベースで増加するものを指す。

2. 準備

本節では、「モジュラリティが変化するパターン」枚挙へ向けての準備について述べる。

連絡先: 原口 誠
北海道大学大学院情報科学研究科
〒 060-0814　札幌市北区北 14 条西 9 丁目
TEL:011-706-7106
E-Mail:mh@ist.hokudai.ac.jp

2.1 アイテムの 1 部グラフ (データ) の定義

まずはじめに、アイテムをノード、アイテム間の共起頻度を重みつきエッジとする 1 部グラフの定義をする。アイテムの集合 $I = \{o_1, o_2, ..., o_m\}$ とし、非空のこの集合をアイテム集合と呼ぶ。また、このアイテム集合の部分集合を要素に持つ集合 $T = \{t_1, t_2, ..., t_n\}$、$t_i \subseteq I$ を個体集合と呼ぶ。そして、個体を行、アイテムを列に持つ個体-アイテム行列を $D = [D_{ij}]$ とし、それを以下に示す。

$$D = \begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_n \end{array} \begin{array}{c} \begin{array}{cccc} o_1 & o_2 & \cdots & o_m \end{array} \\ \begin{pmatrix} D_{11} & D_{12} & \cdots & D_{1m} \\ D_{21} & D_{22} & \cdots & D_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ D_{n1} & D_{n2} & \cdots & D_{nm} \end{pmatrix} \end{array} \quad (1)$$

次に、アイテム間の結びつきを表現する行列 A を定義する。この行列はネットワークの隣接行列に相当するものであり、行列 D の自己相関行列 $D^T D$ として定義する。この行列 A の各要素は行列 D の各列ベクトル (アイテムベクトル) 同士の内積によって、以下のように求められる。

$$\begin{aligned} A &= D^T D \\ &= \begin{pmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{n1} & \cdots & A_{nn} \end{pmatrix} \end{aligned} \quad (2)$$

言い換えると、行列の各要素はアイテム間の共起頻度を示しており、2 つの異なるアイテム間において、多くの個体で共起が見られるとその要素の値は大きくなる。この行列 A によって、アイテムをノード、アイテム間の共起頻度を重み付きエッジとするネットワークとして表現することができる。

3. Newman モジュラリティ

本節では、本研究の基盤理論である Newman モジュラリティについて述べる。

3.1 アイテム間のモジュラリティの定義

本研究では前節で述べた個体-アイテム行列の自己相関行列を DB、DB' それぞれに対して生成する。つまり、DB に対応する個体-アイテム行列の自己相関行列を A、DB' に対応する個体-アイテム行列の自己相関行列を A' とする。そして、Newman のモジュラリティによって DB・DB' それぞれにおけるアイテム間の結び付きを評価する。DB・DB' それぞれのアイテム i 、j 間のモジュラリティは下式のように計算する。

$$A_{ij} - \frac{k_i k_j}{2m}, \tag{3}$$

$$A'_{ij} - \frac{k'_i k'_j}{2m'} \tag{4}$$

ここで、$k_i(k'_i)$ は DB(DB') において構成されたアイテムをノードとするネットワークのノード i の次数、$m(m')$ はノード総数とする。これらは実存のアイテム間の結び付きと次数を期待値の意味で保存するランダムネットワークでの結び付きの差異を観察することによる、アイテム間の結び付きに対する偶発性の離れ具合、すなわちここでの必然性を示している。つまり、「ノード対が結びつく期待値が大きい」ノード対の結び付きを低く評価し、逆に「ノード対が結びつく期待値が小さい」ノード対の結び付きを高く評価する。これにより、出現頻度の低いアイテムの組み合わせ (パターン) に対してもアイテム間のモジュラリティを定義できる。

また、(3)、(4) 式それぞれを要素に持つ DB・DB' のモジュラリティ行列はそれぞれ (5)、(6) 式のように求められる。

$$\begin{pmatrix} A_{11} - \frac{k_1 k_1}{2m} & \cdots & A_{1n} - \frac{k_1 k_n}{2m} \\ \vdots & \ddots & \vdots \\ A_{n1} - \frac{k_n k_1}{2m} & \cdots & A_{nn} - \frac{k_n k_n}{2m} \end{pmatrix} \tag{5}$$

$$\begin{pmatrix} A'_{11} - \frac{k'_1 k'_1}{2m'} & \cdots & A'_{1n} - \frac{k'_1 k'_n}{2m'} \\ \vdots & \ddots & \vdots \\ A_{n1} - \frac{k_n k_1}{2m'} & \cdots & A_{nn} - \frac{k_n k_n}{2m'} \end{pmatrix} \tag{6}$$

4. ネットワーク分割の評価関数

本節では Newman の全ネットワークのグラフ分割の評価関数について述べる。Newman は分割を評価する関数を定め、それを固有値分解に基づく近似を与えた。本研究の目標はクラスタリングではなく、モジュラリティが増加するパターンを検出することであり、単一の順最適なパターンだけでなく、可能性のあるパターンを枚挙することであるが、Newman の分割の評価近似がグループ (パターン) に対する評価の和によって求められている。本研究ではこれに着目し、これをパターンに対する評価式として用いる。

4.1 グループの評価

Newman の全ネットワークのグラフ分割の評価関数は

> そのグループに含まれるノード対のモジュラリティの差異によるエッジ重みの総和をグループの評価とした上で、その総和を全ネットワークの分割評価

として計算することができる。よって、ネットワーク全体のクラスタリングの評価式は、以下のようになる。

$$Q = \sum_{k=1}^{c} \mathbf{s}_k \mathbf{B} \mathbf{s}_k = \mathrm{Tr}(\mathbf{S}^T \mathbf{B} \mathbf{S}) \tag{7}$$

$$\text{ここで、} S_{i,j} = \begin{cases} 1 & (i \in k) \\ 0 & \text{上記以外} \end{cases} \tag{8}$$

つまり、$\mathbf{S}^T\mathbf{BS}$ のトレースをとっていることから、クラスタリングはクラスタ内でのノード対の評価の総和が高くなるように行列 $\mathbf{S}$ の要素の値を決めるという手法で、近似的にクラスタを求める。

4.2 行列の固有値分解と成分の抽出

(5) 式 (または (6) 式) の行列を $\mathbf{B}$ を固有値分解することを考える。(7) 式の 2 次形式は正と負の成分に分かれる。つまり、「実際の結びつきとランダムネットワークでの結びつき」という意味での特徴を持った成分を固有ベクトル得ることができる。

行列 $\mathbf{B}$ の固有値分解は以下のようになる。

$$\mathbf{B} = \mathbf{U}\mathbf{D}\mathbf{U}^{\mathbf{T}} \tag{9}$$

ここで、$\mathbf{U}$ は固有値 β_i に対応する固有ベクトル $\mathbf{u}_i$を列ベクトルとして持つ行列、($n \times 1$ の列ベクトル)、$\mathbf{D}$ は $\beta_1 \geq \beta_2 \geq \cdots \geq \beta_n$ を対角要素に持つ正方行列とする。

4.3 負の成分による制約付きの評価関数

この評価式の最大化を近似するには、正の成分での類似性が高く、かつ負の成分での類似性が低くなるようにノード (ベクトル) を併合していかなくてはならない。正の成分は Newman モジュラリティにおける実存のアイテム間の繋がりの特徴を示し、また負の成分はランダムネットワークにおけるアイテム間の繋がりの特徴を示している。

p、q をそれぞれ選択した正・負の成分数 ($p+q < n$)、$[\mathbf{x}_i]_j = \sqrt{\beta_j} U_{ij}$ をノード i の正成分上でのノードベクトル ($p \times 1$ のベクトル)、$[\mathbf{y}_i]_j = \sqrt{-\beta_{n+1-j}} U_{i,n+1-j}$ をノード i の負成分上でのノードベクトル ($q \times 1$ のベクトル) とする。すると評価式 Q は以下のように書ける。

$$Q = \sum_{k=1}^{c} \left(|\mathbf{X}_k|^2 - |\mathbf{Y}_k|^2 \right) \tag{10}$$

ここで、

$$\mathbf{X}_k = \sum_{i \in G_k} \mathbf{x}_i \text{、} \mathbf{Y}_k = \sum_{i \in G_k} \mathbf{y}_i \tag{11}$$

とした。この評価式でクラスタリングをする場合、単純に正の成分上での類似性のみに着目するだけでなく、負の成分での類似性にも着目する。つまり、類似したノードベクトル同士を併合していくことでグループベクトルが大きくなることから、

> 正のノードベクトル同士で類似し、負のノードベクトル同士では類似しないようなノードベクトル同士の組み合わせを併合する

ことによって、評価式 (7) の最大化を近似する。したがって、クラスタリングの評価関数として (11) 式を用いる場合、$\sum_{k=1}^{c}$ の中にある第 2 項は負の成分による制約項として扱うこととなる。

5. モジュラリティの差異

次に、DB・DB' 間のアイテム間の結びつきのモジュラリティの差異を求める。DB・DB' のモジュラリティの差異を求めるには、「(DB' のモジュラリティ)-(DB のモジュラリティ)((6) 式-(5) 式)」で定義する B_{ij} を下記の式で計算すればよい。

$$B_{ij} = \left(A'_{ij} - \frac{k'_i k'_j}{2m'} \right) - \left(A_{ij} - \frac{k_i k_j}{2m} \right) \tag{12}$$

また、(12) 式を要素に持つ行列を改めて $\mathbf{B}$ とする。

この行列は対称行列であり、4.2 節で述べたモジュラリティ行列の固有値分解と同様の操作をこの行列に適用することで、

> DB・DB' を比較した場合における、DB' のみに見られる高いモジュラリティを特徴として持つ、互いに直交する正の成分と、それを阻害する負の成分

を得ることができる。そして、アイテムを「正 (負) の成分から成る空間上における点 (もしくは、その空間上の原点からその点へ向けたベクトル)」として表現することができる。k-means アルゴリズムや貪欲アルゴリズムを適用することでアイテムをクラスタリングすることも可能となる。これに着目し、本研究では DB' でのモジュラリティの大きさを示す正の成分と、その逆の負の成分に対して制約を課したパターン枚挙手法に応用する形をとる (次節)。

6. パターン枚挙手法

本節では前節でのクラスタリングアルゴリズムで用いたヒューリスティクスを探索プログラム上での制約として用いることによる、「必然性が変化する」パターンを枚挙する探索手法を提案する。

6.1 基本戦略

本研究での探索アルゴリズムは深さ優先探索を応用する形をとる。つまり、部分集合をその大きさが小さな部分集合から順に、アイテムを一つずつ汲みあげながら生成する手法をとる。汲みあげる過程で制約を満たさないものは枝刈りを行うことによって、検出ターゲットのパターンを抽出する。以下に、探索の基本的な流れを示す。

1. 探索を行うため、正・負成分によるノードベクトルの対 $(\mathbf{x}_i, \mathbf{y}_i)$ から成るベクトルの集合を改めてノードベクトルのリスト $\{\mathbf{r}_1, \cdots, \mathbf{r}_n\}$ $(\mathbf{r}_i = (\mathbf{x}_i, \mathbf{y}_i))$ とした集合と結果リスト $F = \{\mathbf{r}_1, \cdots, \mathbf{r}_n\}$ を用意し、それを入力とする。

2. ノードベクトルの集合からノードベクトルの対を一つずつ結合し、パターンを生成する。そのために、深さ優先探索により結合するアイテムの組み合わせを探索する。探索の過程で後述する制約を全て満たすものの中から、正の成分に対する評価 (後述) が上位 N 個 (top-N) のアイテム同士 (またはアイテムとパターン) の組み合わせを、結果リスト F に追加する。

3. 2. を再帰的に繰り返し、制約全てを満たす結合がなくたった時点で探索を終了する

上記の探索を実現させるため、探索においては個々のアイテムをノードベクトルとして管理するノードベクトルリスト $\{\mathbf{r}_1, \cdots, \mathbf{r}_n\}$ と、後述する制約を満たしたパターンを管理する結果リストの 2 つのリストを管理する。

6.1.1 モジュラリティの制約と動的順序

(11) 式の各グループの評価をパターンの評価と見立てて、可能性のあるパターンを枚挙することを考える。アイテムを汲み上げる過程において、アイテム追加後のパターンの集中度、すなわち前節のクラスタリングアルゴリズムでのグループベクトルとノードベクトルの余弦類似度の総和の平均 (クラスタ (パターン) 内の分散に相当する量) がある一定以上 (一定以下) であれば結合し、そうでなければ結合を止めるような探索を行う。

よって、探索の過程におけるノードベクトル (アイテム) の結合に対するモジュラリティの制約は以下の 2 つとなる。

結合後の正の成分に対する制約

$$\frac{1}{|F_k|+1}\left(\sum_{i\in F_k}\cos(\mathbf{X}_k,\mathbf{x}_i)+\cos(\mathbf{X}_k,\mathbf{x}_l)\right) > T_p \tag{13}$$

結合後の負の成分に対する制約

$$\frac{1}{|F_k|+1}\left(\sum_{i\in F_k}\cos(\mathbf{Y}_k,\mathbf{y}_i)+\cos(\mathbf{Y}_k,\mathbf{y}_l)\right) < T_n(=1-T_p) \tag{14}$$

ここで、$|F_k|$ はパターン k に含まれるアイテム数、$\mathbf{x}_l(\mathbf{y}_l)$ は新たに結合するノードベクトルである。次に、(13) 式についての動的順序付けを行い、評価の高いものから上位 N 個 (top-N) のものに対して結合を試みる。アイテムの結合を繰り返すと、(13) 式はすでに得られたクラスタ (パターン) の平均に近い順序でノードベクトル (アイテム) を追加することになることから、評価は単調に悪くなる。よって、正の成分に対する制約下で安全な枝刈りができる。

また、(13) 式の評価の上位 N 個のものについてのみ結合をすることで、パターンの生成過程の途中でも、(13) 式の評価が top-N よりも悪くなった時点で、枝刈りが可能となる。

6.1.2 結合に対する支持度の制約

本研究で検出したいパターンは上記のモジュラリティに関する制約を全て満たし、かつ「DB・DB' 共において頻度が低いもの」である。そこで、頻度の差が大きくなるようなノードベクトル (アイテム) 同士の結合を排除する。よって、結合によって得られるパターンの支持度がある区間内に収まるようにするため、以下のような制約をつける。

結合後の支持度に対する制約

$$\text{DB・DB}'\text{において、}\delta_l \le sup(\{\mathbf{r}_1, \cdots, \mathbf{r}_l\}) \le \delta_h \tag{15}$$

ここで、支持度の制約に下限を設けたのはあまりに出現頻度が低いパターン (例えば、DB・DB' 共において 1 回しか出現しないパターン等) の検出を避けるためである。

6.2 アルゴリズムの詳細

次に、提案手法を実現するアルゴリズムの詳細を示す。

- 入力：ノードベクトルのリスト $\{\mathbf{r}_1, \cdots, \mathbf{r}_n\}(1 \le i \le n$、$\mathbf{r}_i = (\mathbf{x}_i, \mathbf{y}_i))$、結果リスト $List = \{\emptyset\}$、T_p、T_n、δ_l、δ_h、top-N の N
- ノードベクトルのリスト $\{\mathbf{r}_1, \cdots, \mathbf{r}_n\}$ 全てに対して、以下の操作を行う
  - (13)〜(15) 式による制約すべてを満たすもののみ結合し、$List$ にそのパターンを追加する一方、制約を一つでも満たさない結合を禁止する
    - このとき、制約を満たすものの中で、(13) 式の評価が最も高い N 個を選び出し、それらを評価が大きい順に順序付けする

– 制約すべてを満たすノードベクトルの組み合わせが無くなるまで、上記の結合操作を再帰的に繰り返す。

- 出力：結果リスト $List$

7. 実験と考察

本章では、前章で述べたアルゴリズムを評価するために行った実験について述べる。なお、アルゴリズムの実装は Java にて行い、実験環境は Intel Xeon E5520 (2.27GHz)、主記憶 24GB にて行った。実験データは毎日新聞の 1994 年の新聞記事 (DB) と 1995 年の新聞記事 (DB') のうち、「神戸」に関連する記事をそれぞれ用いた。DB での文書数は 2337、DB' での文書数は 9324 であった。

7.1 文書-品詞行列の生成と自己相関行列の生成

DB・DB ' の各文書集合に対して、形態素解析を施し品詞に分解し、名詞のみを抽出した。そして、DB・DB ' それぞれの文書集合を行列で表現するために、tf 値を要素に持つ文書-品詞行列を生成した。文書 t_i に出現する品詞 o_j の tf 値 tf_{ij} は以下の式で求めた。

$$tf_{ij} = \frac{\text{文書 } t_i \text{中における品詞 } o_j \text{の出現回数}}{\text{文書 } t_i \text{中における全品詞数}} \tag{16}$$

この行列から 2.1 節で述べた要領で、DB・DB' それぞれの品詞の共起頻度を要素に持つ自己相関行列を生成した。

7.2 実行時間について

$T_p = 0.9$、$T_n = 0.1$、$\delta_l = 0.001$、$\delta_h = 0.01$、$N = 10$ としたときに、データ数を増加させると実行時間がどのように増加するかを調べた。すると、以下のようなグラフが得られた。上

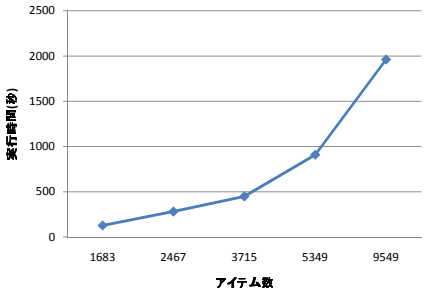

図 1: データ数を増やした時の実行時間の変遷

記の図より、指数関数的に実行時間が増加していることが分かる。これは、アルゴリズムの性質上、探索木の深さに比例して、実行時間も指数関数的に増加するためと考えられる。

7.3 得られたパターンについて

$T_p = 0.9$、$T_n = 0.1$、$\delta_l = 0.001$、$\delta_h = 0.01$、$N = 10$ とし、品詞数 1683 のデータ (7.1 節の実験で用いたデータ) を用いてアルゴリズムを実行すると表 1 のようなパターンが得られた。ちなみに、得られたパターン総数は 3961、実行時間は約 98 秒であった。

得られたパターンは表 1 の反映している文書の内容から見てとれるように、阪神大震災をきっかけとして結び付きに必然性 (モジュラリティ) が高まったと思われる品詞によるパターンが多く検出されることが確認できた。さらに、パターン「決定、発生、被害」等のようにモジュラリティの意味での結合増加が, サポート的には微増のものに対しても起こりえることも確認できた。これは、ある有意なカテゴリ・トピックを示すようなパターンになりつつあるもの、すなわち「ある特定の使われ方」をするようになったものが検出されたためと考えられる。

表 1:検出されたパターンの 1 部 (モジュラリティの差が大きい上位 10 個) と、それらが反映している DB' での文書の概要

得られたパターン	モジュラリティの差	支持度の差	反映していた(DB'での)文書の概要
J R,次,地震	0.0002465	0.0068673	地震による被害を受けた、各種交通機関の復旧
伊丹,南,明石,西宮,須磨,中央	0.0002072	0.0002178	
伊丹,南,明石,宝塚,尼崎	0.0001887	0.001397556	
決定,発生,被害	0.0000590	0.0006479	地震による被害と、その地域の再開発計画の決定
家,代表,生活,住宅	0.0000535	0.0009686	ボランティア代表者の被災者の生活の調査
滋賀,明石,宝塚	0.0000448	0.0037571	各種交通機関の復旧
システム,全国,被害	0.0000439	0.0030063	地震による被害によって表面化した、行政システムの問題
南,明石,宝塚	0.0000436	0.0022567	各種交通機関の復旧
伊丹,南,明石	0.0000430	0.001507	各種交通機関の復旧
センター,会場,中央	0.0000422	0.0037571	チャリティーコンサートの会場案内

8. 今後の課題

本研究では、DB・DB' 共において頻度が低いが、DB' でのみモジュラリティが高いようなパターンの抽出手法を提案した。今後の課題として、大きく 2 つが挙げられる。一つは計算コストの面から、もう一つはモジュラリティの制約の面からの課題である。

8.1 既存アルゴリズムの改良

本稿でのアルゴリズムは素朴に深さ優先探索に対して制約を追加・変更した上で、(13) に対する動的順序をつけた形となっており、結果リストの更新も down closure 法や振り分け技法などを応用して工夫する必要がある。このようにすることで、ある程度、計算コストを軽減できることが考えられる。

8.2 モジュラリティの制約面について

本稿で提案したアルゴリズムでは、負の成分上でのノードベクトル (アイテム) の結合制約 ((14) 式) を設け、それとその他の制約を満たすものに対して順次結合を繰り返すといった、消極的な負の成分の使い方をした。しかし、新たなノードベクトル (アイテム) を結合する際、どの結合候補ともバランスが取れない、すなわち、どの結合候補とも負の成分上でのノードベクトルが集中してしまうものは結合候補から外すといったような負の成分の積極的な使い方もあると考えられる。これによる枝刈りは、正の成分による動的順序の構成の仕方によっては、有効な場合もあると考えられる。また、これにより計算コストもある程度軽減できるのではないかと著者らは考えている。これについては、稿を改めて報告したい。

参考文献

[1] M.E.J.Newman, Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74, 036104 (2006).

[2] 宇野　毅明, 有村　博紀, データインテンシブコンピューティング, その 2 ,-頻出アイテム集合発見アルゴリズム-, 人工知能学会誌,17(2),(2007)

[3] Guozhu Dong and Jinyan Li, Efficient Mining of Emerging Patterns: Discovering Trends and Differences, KDD '99 Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining(1999)